

mu399的专栏

海阔凭鱼跃，天高任鸟飞

≡ 目录视图

≡ 摘要视图

RSS 订阅

个人资料



从杰

访问：295186次

积分：2833

等级：

BLOG > 5

排名：第9313名

原创：45篇 转载：11篇

译文：1篇 评论：134条

文章搜索

Q

文章分类

- 01背包 (1)
- 动态规划 (1)
- 观点 (1)
- 样式 (1)
- 算法 (3)
- 函数式语言 (1)
- 数学 (1)
- 突然间的自我 (1)
- 生活 (1)

文章存档

- 2016年03月 (4)
- 2016年02月 (3)
- 2015年03月 (1)
- 2015年02月 (1)
- 2014年12月 (4)

展开

阅读排行

- 动态规划之01背包问题 | (93311)
- ffmpeg+sdl教程-----练 (16658)
- ffmpeg+sdl教程----编写一 (16038)
- Android menu默认样式的 (11500)
- 简单易懂的Dancing links

【1024程序员节】参加活动领暖心礼品

【观点】有了深度学习，你还学传统机器学习算法么？

【资源库】火爆了的React Native都在研究什么

动态规划之01背包问题（最易理解的讲解）

标签： 算法 c

2012-07-06 17:09 93364人阅读 评论(32) 收藏 举报

≡ 分类： 01背包 动态规划

| 版权声明：本文为博主原创文章，未经博主允许不得转载。

01背包问题，是用来介绍动态规划算法最经典的例子，网上关于01背包问题的讲解也很多，我写这篇文章力争做到用最简单的方式，最少的公式把01背包问题讲解透彻。

01背包的状态转换方程 $f[i,j] = \text{Max}\{ f[i-1,j-W_i]+P_i(j \geq W_i), f[i-1,j] \}$

$f[i,j]$ 表示在前*i*件物品中选择若干件放在承重为 *j* 的背包中，可以取得的最大价值。

P_i 表示第*i*件物品的价值。

决策：为了背包中物品总价值最大化，第 *i*件物品应该放入背包中吗？

题目描述：

有编号分别为a,b,c,d,e的五件物品，它们的重量分别是2,2,6,5,4，它们的价值分别是6,3,5,4,6，现在给你个承重为10的背包，如何让背包里装入的物品具有最大的价值总和？

name	weight	value	1	2	3	4	5	6	7	8	9	10
a	2	6	0	6	6	9	9	12	12	15	15	15
b	2	3	0	3	3	6	6	9	9	9	10	11
c	6	5	0	0	0	6	6	6	6	6	10	11
d	5	4	0	0	0	6	6	6	6	6	10	10
e	4	6	0	0	0	6	6	6	6	6	6	6

只要你能通过找规律手工填写出上面这张表就算理解了01背包的动态规划算法。

首先要明确这张表是至底向上，从左到右生成的。

为了叙述方便，用e2单元格表示e行2列的单元格，这个单元格的意义是用来表示只有物品e时，有个承重为2的背包，那么这个背包的最大价值是0，因为e物品的重量是4，背包装不了。

对于d2单元格，表示只有物品e，d时,承重为2的背包,所能装入的最大价值，仍然是0，因为物品e,d都不是这个背包能装的。

同理，c2=0，b2=3,a2=6。

对于承重为8的背包，a8=15,是怎么得出的呢？

根据01背包的状态转换方程，需要考察两个值，

一个是 $f[i-1,j]$,对于这个例子来说就是b8的值9，另一个是 $f[i-1,j-W_i]+P_i$ ；

[ffmpeg+sdi教程----编写-- \(11147\)](#)
[opencv仿射变换与透视变 \(10816\)](#)
[ffmpeg+sdi教程----编写-- \(10712\)](#)
[Android纯代码实现九宫格 \(10231\)](#)
[Android纯代码实现九宫格 \(10218\)](#)
[ffmpeg+sdi教程----编写-- \(9976\)](#)

评论排行

[动态规划之01背包问题 \(32\)](#)
[ffmpeg+sdi教程----编写-- \(14\)](#)
[ffmpeg+sdi教程-----编 \(13\)](#)
[Android纯代码实现九宫格 \(9\)](#)
[简单易懂的Dancing links \(8\)](#)
[opencv仿射变换与透视变 \(7\)](#)
[ffmpeg+sdi教程----编写-- \(6\)](#)
[ffmpeg+sdi教程----编写-- \(6\)](#)
[FLEX 渲染器的简单使用 \(5\)](#)
[ffmpeg+sdi教程----编写-- \(5\)](#)

推荐文章

* 2016 年最受欢迎的编程语言是什么?
* Chromium扩展 (Extension) 的页面 (Page) 加载过程分析
* Android Studio 2.2 来啦
* 手把手教你做音乐播放器 (二) 技术原理与框架设计
* JVM 性能调优实战之: 使用阿里开源工具 TPProfiler 在海量业务代码中精确定位性能代码

最新评论

[动态规划之01背包问题 \(最易理](#)
12期张欢: 希望博主可以多写点注释, 帮助我这种小菜更好的理解。
[动态规划之01背包问题 \(最易理](#)
ytwu_ucas: 这个图画确实很关键! !
[动态规划之01背包问题 \(最易理](#)
12期-赵尽朝: 图一下子明白了, 自己写了一遍突然明白了
[动态规划之01背包问题 \(最易理](#)
编程与论文stepbystep: 你的背包问题中每类物品最多只能装一个。若能装多个, 求解结果与递推公式要改变
[动态规划之01背包问题 \(最易理](#)
萤火虫小Q: {
answer.Add(item.Name); ...
[动态规划之01背包问题 \(最易理](#)
萤火虫小Q: else if
(bagItems.Weight mi_wi ? mi_1 : mi_wi; ...
[动态规划之01背包问题 \(最易理](#)
萤火虫小Q: public List
GetZeroOnePackAnswer(List
bagItems, in...
[动态规划之01背包问题 \(最易理](#)
萤火虫小Q: public void Test() {
var nameArr...
[动态规划之01背包问题 \(最易理](#)
萤火虫小Q: //楼主这代码有问题的哦, 你再看看, 图还是不错的。你说表的构造由下而上, 但是你的代码显示是从上到下的...
[Android纯代码实现九宫格解锁](#)

在这里,

$f[i-1,j]$ 表示我有一个承重为8的背包, 当只有物品b,c,d,e四件可选时, 这个背包能装入的最大价值

$f[i-1,j-W[i]$ 表示我有一个承重为6的背包 (等于当前背包承重减去物品a的重量), 当只有物品b,c,d,e四件可选时, 这个背包能装入的最大价值

$f[i-1,j-W[i]$ 就是指单元格b6,值为9, P指的是a物品的价值, 即6

由于 $f[i-1,j-W[i]+P_i = 9 + 6 = 15$ 大于 $f[i-1,j] = 9$, 所以物品a应该放入承重为8的背包

以下是actionscript3 的代码

```
[java]
01. public function get01PackageAnswer(bagItems:Array,bagSize:int):Array
02. {
03.     var bagMatrix:Array=[];
04.     var i:int;
05.     var item:PackageItem;
06.     for(i=0;i<bagItems.length;i++)
07.     {
08.         bagMatrix[i] = [0];
09.     }
10.     for(i=1;i<=bagSize;i++)
11.     {
12.         for(var j:int=0;j<bagItems.length;j++)
13.         {
14.             item = bagItems[j] as PackageItem;
15.             if(item.weight > i)
16.             {
17.                 //i背包装不下item
18.                 if(j==0)
19.                 {
20.                     bagMatrix[j][i] = 0;
21.                 }
22.                 else
23.                 {
24.                     bagMatrix[j][i]=bagMatrix[j-1][i];
25.                 }
26.             }
27.             else
28.             {
29.                 //将item装入背包后的价值总和
30.                 var itemInBag:int;
31.                 if(j==0)
32.                 {
33.                     bagMatrix[j][i] = item.value;
34.                     continue;
35.                 }
36.                 else
37.                 {
38.                     itemInBag = bagMatrix[j-1][i-item.weight]+item.value;
39.                 }
40.                 bagMatrix[j][i] = (bagMatrix[j-1][i] > itemInBag ? bagMatrix[j-1][i] : itemInBag)
41.             }
42.         }
43.     }
44.     //find answer
45.     var answers:Array=[];
46.     var curSize:int = bagSize;
47.     for(i=bagItems.length-1;i>=0;i--)
48.     {
49.         item = bagItems[i] as PackageItem;
50.         if(curSize==0)
51.         {
52.             break;
53.         }
54.         if(i==0 && curSize > 0)
55.         {
56.             answers.push(item.name);
57.             break;
58.         }
59.         if(bagMatrix[i][curSize]-bagMatrix[i-1][curSize-item.weight]==item.value)
60.         {
61.             answers.push(item.name);
62.             curSize -= item.weight;
```

写代码的火星人: 不够4个点就报错啊09-05 17:26:56.339: E/AndroidRuntime(2213...

```
63.         }
64.     }
65.     return answers;
66. }
```

PackageItem类

```
[java]
01. public class PackageItem
02. {
03.     public var name:String;
04.     public var weight:int;
05.     public var value:int;
06.     public function PackageItem(name:String,weight:int,value:int)
07.     {
08.         this.name = name;
09.         this.weight = weight;
10.         this.value = value;
11.     }
12. }
```

测试代码

```
[java]
01. var nameArr:Array=['a','b','c','d','e'];
02. var weightArr:Array=[2,2,6,5,4];
03. var valueArr:Array=[6,3,5,4,6];
04. var bagItems:Array=[];
05. for(var i:int=0;i<nameArr.length;i++)
06. {
07.     var bagItem:PackageItem = new PackageItem(nameArr[i],weightArr[i],valueArr[i]);
08.     bagItems[i]=bagItem;
09. }
10. var arr:Array = ac.get01PackageAnswer(bagItems,10);
```

顶 20 踩 10

上一篇 最长公共子序列问题的改进算法
下一篇 动态规划之树形动态规划

猜你在找

- | | |
|------------------------------|----------------------|
| 《C语言/C++学习指南》加密解密篇（安全相关算法） | 动态规划之01背包问题讲解 |
| C语言系列之 字符串相关算法 | 从01背包问题理解动态规划 |
| C语言系列之 字符串压缩算法与结构体初探 | 从01背包问题理解动态规划 |
| C语言系列之 递归算法示例与 Windows 趣味小项目 | 数据结构基础 从01背包问题理解动态规划 |
| C语言系列之 数组与算法实战 | 从01背包问题理解动态规划——初体验 |

查看评论

29楼 12期张欢 2016-10-12 16:34发表



希望博主可以多写点注释，帮助我这种小菜更好的理解。

28楼 ytwu_ucas 2016-10-09 16:00发表



这个图画的确很关键！！

27楼 12期-赵尽朝 2016-10-08 15:19发表



图一下子明白了，自己写了一遍突然明白了

26楼 编程与论文stepbystep 2016-09-29 18:03发表



你的背包问题中每类物品最多只能装一个。若能装多个，求解结果与递推公式要改变

25楼 萤火虫小Q 2016-09-07 22:32发表



```
[csharp]
01. //楼主这代码有问题的哦，你再看下，图还是不错的。你说表的构造由下而上，但是你的代码显示的，这个是不对的，可以看看我写的：C#实现。
02. //首先定义一个类用于存储项
03. /// <summary>
04. /// 物品项
05. /// </summary>
06. public sealed class PackageItem
07. {
08.     /// <summary>
09.     /// 物品名称
10.     /// </summary>
11.     public string Name { get; set; }
12.
13.     /// <summary>
14.     /// 物品重量
15.     /// </summary>
16.     public int Weight { get; set; }
17.
18.     /// <summary>
19.     /// 价值
20.     /// </summary>
21.     public int Value { get; set; }
22.
23.     /// <summary>
24.     /// 构造函数
25.     /// </summary>
26.     /// <param name="name"></param>
27.     /// <param name="weight"></param>
28.     /// <param name="value"></param>
29.     public PackageItem(string name, int weight, int value)
30.     {
31.         this.Name = name;
32.         this.Weight = weight;
33.         this.Value = value;
34.     }
35. }
```

Re: 萤火虫小Q 2016-09-07 22:34发表



```
[csharp]
01. public void Test()
02. {
03.     var nameArr = new[] { "a", "b", "c", "d", "e" };
04.     var weightArr = new[] { 2, 2, 6, 5, 4 };
05.     var valueArr = new[] { 6, 3, 5, 4, 6 };
06.     var bagItems = nameArr.Select((t, i) => new PackageItem(t, weightArr[i], valueArr[i]));
07.     var answer = GetZeroOnePackAnswer(bagItems, 10);
08.     answer.ForEach(System.Console.WriteLine);
09. }
10.
11. }
```

Re: 萤火虫小Q 2016-09-07 22:35发表



```
[csharp]
01. else if (bagItems[i].Weight <= j)
02.     {
03.         var mi_1 = bagMatrix[i + 1, j];
04.         var mi_wi = bagMatrix[i + 1, j - bagItems[i].Weight];
05.         bagMatrix[i, j] = mi_1 > mi_wi ? mi_1 : mi_wi;
06.     }
07.     }
08.     }
09.     var answer = new List<string>();
10.     var currentSize = bagSize;
11.     for (var k = 0; k < bagItems.Count; k++)
12.     {
13.         var item = bagItems[k];
14.         if (currentSize == 0)
15.         {
16.             break;
17.         }
18.         if (k == bagItems.Count-1 && currentSize >= item.Weight)
19.         {
20.             answer.Add(item.Name);
21.         }else
22.         if (bagMatrix[k, currentSize] - bagMatrix[k + 1, currentSi
23.
```

Re: 萤火虫小Q 2016-09-07 22:36发表



```
[csharp]
01. {
02.     answer.Add(item.Name);
03.     currentSize -= item.Weight;
04. }
05. }
06. return answer;
07. }
```

Re: 萤火虫小Q 2016-09-07 22:35发表



```
[csharp]
01. public List<string> GetZeroOnePackAnswer(List<PackageItem> bagItems, int t
02.     {
03.         var goodsNumber = bagItems.Count;
04.         var bagMatrix = new int[goodsNumber, bagSize + 1];
05.         for (var j = 0; j <= bagSize; j++)
06.         {
07.             for (int i = goodsNumber - 1; i >= 0; i--)
08.             {
09.                 if (j == 0) bagMatrix[j, i] = 0;
10.                 else if (i == goodsNumber - 1)
11.                 {
12.                     if (bagItems[i].Weight > j) bagMatrix[i, j] = 0;
13.                     else bagMatrix[i, j] = bagItems[i].Value;
14.                 }
15.                 else if (bagItems[i].Weight > j)
16.                 {
17.                     bagMatrix[i, j] = bagMatrix[i + 1, j];
18.                 }
19.
```

24楼 PrayLife 2016-08-19 22:44发表



楼主写的很好！能有Java或者C++的代码就更好了！

23楼 loosen_q 2016-08-06 11:49发表



那张图给的很不错，点赞

22楼 baidu_34951843 2016-05-10 20:36发表



表格边缘初始化没讲清楚啊

21楼 qq_26678273 2016-05-08 19:20发表



abcde物品顺序没影响吗？e2代表有个容量2的背包，只有一个物品e待选。那只有一个容量为2的背包，只有一个物品a待选这种情况怎么表示？不能排除这种情况啊

20楼 一路走来的自己 2016-05-07 21:29发表



看到这终于懂了~~谢谢

19楼 snail_zhous 2015-12-11 09:58发表



虽然说不知道你在说什么，但感觉很牛B的样子，感觉楼主懂的真多，给100个赞

18楼 u010741588 2015-11-09 21:26发表



写的很棒，加上七楼推荐的博文，基本能搞懂动态规划了

17楼 杨晓菲 2015-10-26 14:16发表



看起来很容易懂！

16楼 川师小灰灰 2015-10-22 10:51发表



楼主辛苦了,能不能用c写一遍？

15楼 zzl1277120515 2015-10-18 15:31发表



赞

14楼 郑浩- 2015-10-11 15:33发表



写的非常的好了，谢谢

13楼 qq_21311539 2015-08-13 15:13发表



觉得7楼提供的那个地方讲解的更加仔细，对于dp一些要注意的地方都讲了

12楼 Divine- 2015-06-16 21:53发表



确实很给力 赞一个

11楼 秦岭脚下的苦行僧 2015-03-25 17:24发表



```
//find answer 中
if(i==0 && curSize > 0)
    此处是不对的，应该是
    if(i==0 && curSize > =item.weight)
    因为最优解可能不包含第一个物品，同时背包也没有装满，若只是判断大于0，就会误将第一个物品也算入背包中
```

10楼 李博Garvin 2015-02-04 10:54发表



赞，学习下

9楼 xu456as 2015-01-21 20:10发表



例子太形象了，美中不足的是表中有些数据是错的，还是感谢楼主！

8楼 nxyb 2015-01-07 11:22发表



你给出的表和 $f[i,j] = \text{Max}\{f[i-1,j-W_i] + P_i(j \geq W_i), f[i-1,j]\}$ 对不上

7楼 CheatDeath2142 2014-12-06 07:36发表



博主写的很不错，再加上<http://www.cnblogs.com/SDJL/archive/2008/08/22/1274312.html>的文章就能更好理解背包问题了

6楼 Morph_Zhou 2014-11-11 11:49发表



棒

5楼 C洛克 2014-11-09 12:52发表



给力！！赞一个

4楼 L136423634 2014-10-08 16:25发表



感谢分享！

3楼 cugbin 2014-07-26 13:11发表



赞一个！

2楼 xidianjunnan 2014-07-05 11:06发表



讲的太好了，代码用c/c++写就更好了

1楼 DongKhan 2014-05-24 12:07发表



写的很通俗易懂，对我这种一看到公式就眼晕的人来说帮助很大。谢谢楼主，赞一个！

您还没有登录,请[登录](#)或[注册](#)

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

核心技术类目

全部主题 Hadoop AWS 移动游戏 Java Android iOS Swift 智能硬件 Docker OpenStack
VPN Spark ERP IE10 Eclipse CRM JavaScript 数据库 Ubuntu NFC WAP jQuery
BI HTML5 Spring Apache .NET API HTML SDK IIS Fedora XML LBS Unity
Splashtop UML components Windows Mobile Rails QEMU KDE Cassandra CloudStack FTC
coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo
Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase Pure Solr
Angular Cloud Foundry Redis Scala Django Bootstrap

公司简介 | 招贤纳士 | 广告服务 | 银行汇款帐号 | 联系方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

网站客服 杂志客服 微博客服 webmaster@csdn.net 400-600-2320 | 北京创新乐知信息技术有限公司 版权所有 | 江苏乐知网络技术有限公司 提供商务支持

京 ICP 证 09002463 号 | Copyright © 1999-2016, CSDN.NET, All Rights Reserved

